APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/963,783	09/27/2001	Tadashi Shimoji	0035/019001	9152

22893   7590   01/13/2005

SMITH PATENT OFFICE
1901 PENNSYLVANIA AVENUE N W
SUITE 200
WASHINGTON, DC 20006

EXAMINER
WON, MICHAEL YOUNG

ART UNIT	PAPER NUMBER
2155	

DATE MAILED: 01/13/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE <u>3</u> MONTH(S) FROM
THE MAILING DATE OF THIS COMMUNICATION.
- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed
  after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
  Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any
  earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on <u>27 September 2001</u>.

2a)☐ This action is **FINAL.**   2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is
closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) <u>1-29</u> is/are pending in the application.

    4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) <u>1-29</u> is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

    Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

    Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

    a)☐ All b)☐ Some * c)☒ None of:

      1.☒ Certified copies of the priority documents have been received.

      2.☐ Certified copies of the priority documents have been received in Application No. _____.

      3.☐ Copies of the certified copies of the priority documents have been received in this National Stage
application from the International Bureau (PCT Rule 17.2(a)).

    * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1)☒ Notice of References Cited (PTO-892)

2)☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3)☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
    Paper No(s)/Mail Date _____.

4)☐ Interview Summary (PTO-413)
    Paper No(s)/Mail Date. _____ .

5)☐ Notice of Informal Patent Application (PTO-152)

6)☐ Other: _____.

DETAILED ACTION

1.   Claims 1-29 have been examined and are pending with this action.

Claim Rejections - 35 USC § 112

The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter, which the applicant regards as his invention.

2.   Claims 1-29 rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. Although a broad description is given for each module, the claim language lacks any combinational feature to teach one of ordinary skill in the art how all the modules are employed as a whole to result in the intended invention. In other words, the examiner could not conclude the relevance of each module to another module(s) to derive at the conclusion of performing the intended functionality as claimed.

Although the claims are interpreted in light of the specification, limitations from the specification are not read into the claims. See *In re Van Geuns*, 988 F.2d 1181, 26 USPQ2d 1057 (Fed. Cir. 1993).

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all

obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set
forth in section 102 of this title, if the differences between the subject matter sought to be patented and
the prior art are such that the subject matter as a whole would have been obvious at the time the
invention was made to a person having ordinary skill in the art to which said subject matter pertains.
Patentability shall not be negatived by the manner in which the invention was made.

3.   Claims 1-29 are rejected under 35 U.S.C. 103(a) as being unpatentable over

Yokote (US 6,105,074 A).

As per *claim 1*, Yokote teaches a system for dynamically generating and

processing a program (see col.2, lines 13-20) by connecting a server computer (see

col.3, lines 9-10) and at least one of a client computer (see col.3, lines 9-10) and a data

processing server computer (see col.3, lines 9-10) via a network means (see col.3, lines

9-10), sending and receiving data there between, and executing a desired voluntary

data processing process by dynamically generating and then processing at least one

unit-program for data processing (see col.17, lines 13-21), said system comprising: a

functional module storage means for storing a plurality of functional module classes,

wherein each of said functional module classes (see col.13, line 64-col.14, line 3) has a

coded processing logic for processing at least a portion of the unit-program (see col.1,

lines 14-20); a configuration information storage means for storing a plurality of

configuration information including at least request information to read out at least one

of the functional module classes and a processing condition (implicit: see col.18, lines

33-37); a definition information input means for inputting at least one definition

information to declare the contents of a data processing process to be executed (see

col.10, lines 45-54); a configuration information read-out means for reading out at least

one of the configuration information corresponding to said at least one of the definition

information from said configuration information storage means (implicit: see col.18, lines

33-37); a unit-program generating means for reading out at least one of the functional

module classes corresponding to said at least one of the configuration information from

said functional module storage means, wherein said unit-program generating means

dynamically generates a unit-program by using the coded processing logic from said

functional module classes (see col.2, lines 13-20 and col.17, lines 13-21); and a unit-

program processing means for dynamically executing said unit-program by using said

processing condition included in said configuration information (see col.17, lines 13-21).

As per _claim 2_, Yokote teaches of further comprising a configuration information

request means for requesting at least one of the configuration information for executing

the data processing, said configuration information storage means for storing the to the

configuration information corresponding data processing to be executed, wherein said

configuration information have been the used for generating the data of the unit-

program, said configuration information read-out means reads out the configuration

information from said configuration information storage means based on the request

from said configuration information request means (implicit: see col.18, lines 18-22).

As per _claim 3_, Yokote teaches a system for dynamically generating and

processing a program by connecting a server computer and a client computer and a

data processing server computer via a network means, sending and receiving data

there between, and executing the desired voluntary data processing process by

dynamically generating and processing at least one unit-program for data processing,

said system comprising: a functional module storage means for storing a plurality of

functional module classes, wherein each of said functional module classes has a coded

processing logic for processing at least a portion of the unit-program (see claim 1

rejection above); a configuration information storage means for storing a plurality of

configuration information corresponding to each of a plurality of data processing

processes, wherein said configuration information includes at least request information

to read out least one of the functional module classes and a processing condition (see

claim 1 rejection above); a configuration information request means for requesting at

least one of the configuration information for executing the data processing process

(see claim 2 rejection above); a configuration information read-out means for reading

out at least one of the configuration information corresponding to said request from the

configuration information request means (see claim 2 rejection above); a unit-program

generating means for reading out at least one of the functional module classes

corresponding to said at least one of the configuration information from said functional

module storage means, wherein said unit-program generating means dynamically

generates a unit-program by using the coded processing logic from said functional

module classes (see claim 1 rejection above); and a unit-program processing means for

dynamically executing said unit-program based on said processing condition included in

said configuration information (see claim 1 rejection above).

As per _claim 4_, Yokote further teaches wherein said configuration information

storage means stores least one functional module class having a coded processing

logic for handling at least one a variable data and a parameter (see col.13, lines 1-45),

said definition information input means inputs at least one of definition information to

declare the contents of the data processing process and at least one of the variable

data and the parameter (see col.13, lines 1-45), said configuration information read-out

means reads out at least one of the configuration information from said configuration

storage means corresponding to said least one the definition information and the

request from said configuration information request means (see claim 1 and 2 rejection

above), and said unit-program generating means reads out at least one of the functional

module classes including at least one functional module class from said functional

module storage means corresponding to said at least one the configuration information

(see claim 1 rejection above), wherein the unit-program generating means dynamically

generates the unit-program by using both the coded processing logic from said

functional module classes (see claim 1 rejection above) and said least one variable data

and the parameter included in the configuration information (see col.13, lines 1-45).

As per _claim 5_, Yokote further teaches wherein said server computer comprises

said configuration information storage means and said configuration information read-

out means (see col.18, line 35: "external storage location"), said client computer

comprises said functional module storage means, said unit-program generating means

and said unit-program processing means (see col.17, lines 5-21: API is downloaded and

generation and execution occurs in the client device).

As per _claim 6_, Yokote further teaches wherein said client computer further

comprises said configuration information request means (inherent).

As per *claim 7*, Yokote further teaches wherein said server computer further

comprises said functional module storage means, said unit-program generating means,

said unit-program processing means and a processing result output means which

returns a processing result of the unit-program to at least one of the client computer, the

server computer and the data processing server computer (Yokote teaches that the

client and the server are interchangeable: see Fig.1 and col.3, lines 9-10: "data

processing device").

As per *claim 8*, Yokote further teaches wherein said data processing server

computer comprises said functional module storage means, said unit-program

generating means and said unit-program processing means (Yokote teaches that the

client and the server are interchangeable: see Fig.1 and col.3, lines 9-10: "data

processing device").

As per *claim 9*, Yokote further teaches wherein said definition information

includes information relating a combination of the functional module classes and a

processing order of the functional module classes for executing the data processing

process (see col.6, lines 56-62).

As per *claim 10*, Yokote teaches a client computer (see claim 5 rejection above)

in a system for dynamically generating and processing a program by connecting to a

server computer via a network means, sending and receiving data there between, and

executing a desired voluntary data processing process by dynamically generating and

processing at least one unit-program for a data processing process, said client

computer comprising: a functional module storage means (see claim 5 rejection above:

client comprises such means) for storing a plurality of functional module classes,

wherein each of said functional module classes comprises a coded processing logic for

processing at least a portion of the unit-program (see claim 1 rejection above); a

definition information input means (implicit: see claim 9 rejection above) for inputting at

least one definition information to declare the contents of a data processing process to

be executed (see claim 1 rejection above); a unit-program generating means (see claim

5 rejection above: client comprises such means) for reading out at least one of said

functional module classes corresponding to at least one of the configuration information

from said functional module storage means (see claim 1 rejection above) when said at

least one of the configuration information including at least request information to read

out at least one said functional module classes and a processing condition are sent

from the server computer (see claim 1 rejection above: "configuration information

storage means"; and claim 5 rejection above: server comprises such means), and then

generating a unit-program by using the coded processing logic from said functional

module classes (see claim 1 rejection above); and a unit-program processing means

(see claim 5 rejection above: client comprises such means) for dynamically executing

said unit-program based on said processing condition included in said configuration

information (see claim 1 rejection above).

As per *claim 11*, Yokote further teaches wherein said server computer stores the

configuration information used to generate the unit program corresponding to the data

processing to be executed (see claim 5 rejection above: server comprises such means),

said client computer further comprises a configuration information request means (see

claim 6 rejection above: client comprises such means) for requesting at least one of the

configuration information corresponding to the data processing to be executed (see

claim 2 rejection above).

As per *claim 12*, Yokote teaches a client computer in a system for dynamically

generating and processing a program by connecting to a server computer via a network

means, sending and receiving data there between, and executing a desired voluntary

data processing process by dynamically generating and processing at least one unit-

program for data processing process, said client computer comprising: a functional

module storage means (see claim 5 rejection above: client comprises such means) for

storing a plurality of functional module classes, wherein each of said functional module

classes comprises a coded processing logic for processing at least a portion of the unit-

program (see claim 1 rejection above); a configuration information request means (see

claim 6 rejection above: client comprises such means) for requesting a configuration to

be sent to the client computer corresponding to the data processing to be executed (see

claim 2 rejection above); a unit-program generating means (see claim 5 rejection above:

client comprises such means) for reading out at least one of said functional module

classes corresponding to at least one of the configuration information from said

functional module storage means when said at least one of the configuration information

including at least request information to read out at least one said functional module

classes and a processing condition are sent from the server computer (see claim 10

rejection above), and then generating a unit-program by using the coded processing

logic from said functional module classes (see claim 1 rejection above); and a unit-

program processing means (see claim 5 rejection above: client comprises such means)

for dynamically executing said unit-program based on said processing condition

included in said configuration information (see claim 1 rejection above).

As per _claim 13_, Yokote further teaches wherein said functional module storage

means (see claim 5 rejection above: client comprises such means) stores at least one

functional module class having the coded processing logic for handling least one of a

variable data and a parameter (see claim 1 and 4 rejections above), said definition

information input means (see claim 10 rejection above: client comprises such means)

inputs at least one of definition information to declare the contents of a data processing

process to be executed and at least one of the variable data and the parameter (see

claim 1 and 4 rejections above), and said unit-program generating means (see claim 5

rejection above: client comprises such means) reads out at least one of said functional

module classes including at least one functional module class for handling at least one

of the variable data and the parameter corresponding to said at least one of the

configuration information from said functional module storage means when said

configuration information including at least information relating to the at least one

functional module class based on said definition information or said request for sending

the configuration information are sent from the server computer (see claim 1 rejection

above), and dynamically generating the unit-program by using both of the coded

processing logic from said at least one functional module class and said at least one of

the variable data and the parameter included in the configuration information (see claim

4 rejection above).

As per *claim 14*, Yokote teaches a server computer in a system for dynamically

generating and processing a program by connecting to at least one of a client computer

and a data processing server computer via a network means, sending and receiving

data there between, and making at least one of the client computer and the processing

server computer execute a desired voluntary data processing process by dynamically

generating and processing at least one unit-program for data processing, said server

computer comprising: a configuration information storage means (see claim 5 rejection

above: server comprises such means) for storing a plurality of configuration information

including least request information to read out at least one functional module class and

a processing condition, wherein each the functional module classes comprises a coded

processing logic for processing at least a portion of the unit-program (see claim 1

rejection above); and configuration information read-out means (see claim 5 rejection

above: server comprises such means) for reading out at least one of the configuration

information corresponding to at least one definition information from said configuration

information storage means (see claim 1 rejection above) when said definition

information declares the contents of a data processing process to be executed is sent

from the client computer (see claim 1 rejection above: "definition information input

means"; and claim 10 rejection above: client computer comprises such means), sending

and providing said read-out configuration information to at least one the client computer

(see claim 1 rejection above: "unit program generating means"; and claim 5 rejection

above: client comprises such means) and the data processing server computer (see

claim 7 rejection above: data processing server computer comprises such means),

whereby at least one of the client computer and the data processing server computer dynamically generates and processes at least one unit-program based on the processing condition included in the configuration information (see claim 1 rejection above).

As per _claim 15_, Yokote further teaches wherein said configuration information storage means (see claim 5 rejection above: server comprises such means) stores at least one of the configuration information which used for generating said unit-program, corresponding the data processing (see claim 2 rejection above or claim 3 rejection above), and said configuration information read-out means (see claim 5 rejection above: server comprises such means) reads out least one of the configuration information corresponding said request for the configuration information (see claim 2 rejection above: "configuration information request means") sent from said client computer (see claim 6 rejection above: client comprises such request means).

As per _claim 16_, Yokote teaches a server computer in a system for dynamically generating and processing a program by connecting to at least one of a client computer and a data processing server computer via network a means, sending and receiving data there between, and making least one of the client computer and the data processing server computer execute a desired voluntary data processing process by dynamically generating and processing at least one unit-program for data processing, said server computer comprising: a configuration information read-out means (see claim 5 rejection above: server comprises such means) for reading out at least one of configuration information corresponding to a request which corresponds to a data

processing be executed from a configuration information storage means (see claim 1

rejection above) when said request to read out the configuration information is sent from

the client computer (see claim 2 rejection above: "configuration information request

means"; and claim 6: client comprises such means), sending and providing said read-

out configuration information to at least one of the client computer (see claim 1 rejection

above: "unit program generating means"; and claim 5 rejection above: client comprises

such means) and the data processing server computer (see claim 7 rejection above:

data processing server computer comprises such means), whereby at least one of the

client computer and the data processing server computer dynamically generates and

processes said unit-program based on a processing condition included in the

configuration information (see claim 1 rejection above).

As per _claim 17_, Yokote further teaches wherein said server computer further

comprises: a configuration information storage means (see claim 5 rejection above:

server comprises such means) for storing plurality of configuration information coding

the coded processing logic for processing at least a portion of the unit-program (see

claim 1 rejection above); a unit-program generating means (see claim 7 rejection above:

server comprises such means) for reading out at least one of said functional module

classes corresponding to the definition information from said functional module storage

means (see claim 1 rejection above) when said definition information for declaring the

contents the data processing process to be executed are sent from said client computer

(see claim 1 rejection above: "definition information input means"; and claim 10 rejection

above), wherein said unit-program generating means dynamically generates the unit-

program by using the coded processing logic from said functional module classes (see

claim 1 rejection above); a unit-program processing means (see claim 7 rejection

above: server comprises such means) for dynamically executing said unit-program

based on the processing condition included in said configuration information (see claim

1 rejection above); and a processing result output means for returning a processing

result of the unit-program to at least one of the client computer and the data processing

server computer (see claim 7 rejection above).

As per _claims 18_, Yokote teaches a method for dynamically generating and

processing a program by connecting a server computer and at least one of a client

computer and a data processing server computer via a network means, sending and

receiving data there between, and executing a desired voluntary data processing

process by dynamically generating and processing at least one unit-program for data

processing in at least one of the client computer and the data processing server

computer, said method comprising the steps of: storing a plurality of functional module

classes into functional module storage means and storing a plurality of configuration

information into a configuration information storage means, wherein each of said

functional module classes comprises a coded processing logic for processing at least a

portion of a unit-program processing and said configuration information includes at least

request information to read out at least one of the functional module classes and a

processing condition; inputting at least one definition information to declare the contents

of a data processing to be executed via a definition information input means; reading

out at least one the configuration information corresponding to said at least one of the

definition information from said configuration information storage means via

configuration information read-out means; reading out at least one of the functional

module classes corresponding to said at least one of the configuration information from

said functional module storage means via a unit-program generating means, and

dynamically generating the unit-program processing by using the coded processing

logic from said functional module classes via said unit-program generating means; and

dynamically executing said unit-program of the data processing based on the

processing condition included in said configuration information via a unit-program

processing means (see claim 1 rejection above).

As per _claim 19_, Yokote further teaches wherein said method further comprises

the steps of: storing the configuration information corresponding to the data processing

to be executed into said configuration information storage means wherein said

configuration information is used for generating the data of the unit-program, requesting

at least one of the for used configuration information for executing the data processing

via a configuration information request means, and reading out the configuration

information from said configuration information storage means based on the request of

said configuration information request means via the configuration information read-out

means (see claim 2 rejection above).

As per _claims 20_, Yokote teaches a method for dynamically generating and

processing a program by connecting a server computer and at least one of a client

computer and a data processing server computer via a network means, sending and

receiving data there between, and executing a desired voluntary data processing

process by dynamically generating and processing at least one unit-program for data processing in at least one of the client computer and the data processing server computer, said method comprising the steps of: storing a plurality of functional module classes into functional module storage means and storing a plurality of configuration information into a configuration information storage means, wherein each of said functional module classes comprises a coded processing logic for processing at least a portion of a unit-program processing and said configuration information includes at least request information to read out at least one of the functional module classes and a processing condition; inputting at least one definition information to declare the contents of a data processing to be executed via a definition information input means; sending the configuration information corresponding to contents of a data processing to be executed via a configuration information request means (see claims 18 and 19 rejections above); reading out at least one the configuration information corresponding to said at least one of the definition information from said configuration information storage means via configuration information read-out means; reading out at least one of the functional module classes corresponding to said at least one of the configuration information from said functional module storage means via a unit-program generating means, and dynamically generating the unit-program processing by using the coded processing logic from said functional module classes via said unit-program generating means; and dynamically executing said unit-program of the data processing based on the processing condition included in said configuration information via a unit-program processing means (see claim 19 rejection above).

As per *claim 21*, Yokote teaches a computer-readable and -recordable media for

controlling at least one of a client computer and a data processing server computer

comprising a system for dynamically generating and processing a program by

connecting a server computer and at least one of the client computer and the data

processing server computer via a network means, sending and receiving data there

between, and executing a desired voluntary data processing process by dynamically

generating and processing at least one unit-program for data processing in at least one

of the client computer and the data processing server computer, said media comprising:

a controlling program for storing a plurality of functional module classes having a coded

processing logic; a controlling program for reading out at least one of said functional

module classes and for dynamically generating a unit-program processing by using the

coded processing logic of said functional module classes; a controlling program for

dynamically executing said unit-program of data processing based on a processing

condition included in said configuration information (see claim 1 and 10-13 rejections

above).

As per *claim 22*, Yokote teaches a computer-readable and -recordable media for

controlling at least one of a client computer and a data processing server computer

comprising a system for dynamically generating and processing a program by

connecting a server computer and at least one of the client computer and the data

processing server computer via a network means, sending and receiving data there

between, and executing a desired voluntary data processing process by dynamically

generating and processing at least one unit-program for data processing in at least one

of the client computer and the data processing server computer, said recordable media comprising: a controlling program for storing a plurality of configuration information including at least one functional module read-out information and a processing condition, wherein a plurality of functional module classes code a logic of a data processing process to be executed; a controlling program for reading out the configuration information and for sending the read-out configuration information to at least one of the client computer and data processing server computer when definition information to declare the contents of the data processing process to be executed is sent from the client computer; a controlling program for storing the configuration information including a read-out information for reading out said functional module classes that code the logic of the data processing; and a controlling program for reading out the configuration information and for sending the read-out configuration information to at least one of the client computer and the data processing server computer when the definition information to declare the contents of the data processing process to be executed are sent from the client computer (see claim 1 and 10-13 rejections above).

As per _claim 23_, Yokote further teaches wherein said media further comprises a controlling program for storing at least one of configuration information corresponding to the data processing to be executed, said configuration information is used for generating a unit-program processing, and a controlling program for reading out at least one of the configuration information based on request information for reading out the configuration information corresponding to the data processing to be executed when said request is sent from the client computer (see claim 2 rejection above).

As per *claims 24*, Yokote teaches a computer-readable and -recordable media

for controlling at least one of a client computer and a data processing server computer

comprising a system for dynamically generating and processing a program by

connecting a server computer and at least one of the client computer and the data

processing server computer via a network means, sending and receiving data there

between, and executing a desired voluntary data processing process by dynamically

generating and processing at least one unit-program for data processing in at least one

of the client computer and the data processing server computer, said media comprising:

a controlling program for storing a plurality of functional module classes having a coded

processing logic; a controlling program for outputting a request of the configuration

information corresponding a data processing to be executed; a controlling program for

reading out at least one of said functional module classes and for dynamically

generating a unit-program processing by using the coded processing logic of said

functional module classes when the configuration information including at least

functional module read-out information and a processing condition a are sent from said

server computer; a controlling program for dynamically executing said unit-program of

data processing based on a processing condition included in said configuration

information (see claim 1 and 10-13 rejections above).

As per *claim 25*, Yokote teaches a computer-readable and -recordable media for

controlling a server computer comprising a system dynamically generating and

processing a program by connecting a server computer and least one of the client

computer and the data processing server computer via a network means, sending and

receiving data there between, and executing a desired voluntary data processing

process by dynamically generating and processing at least one unit-program for data

processing in least one of the client computer and the data processing server computer,

said media comprising: a controlling program for storing configuration information

including at least functional module read-out information corresponding to a data

processing and a processing condition, wherein a plurality functional module classes

code a logic of data processing; and controlling program for reading out the

configuration information and for sending the read-out configuration information to at

least one the client computer and the data processing server computer when a request

for the configuration information corresponding to the data processing to be executed

are sent from the client computer (see claim 1 and 14-16 rejections above).

As per claim 26, Yokote teaches a program transfer system for transferring and

downloading a controlling program to at least one of a client computer and a data

processing server computer comprising a system for dynamically generating and

processing a program by connecting a server computer and at least one of the client

computer and the data processing server computer via a network means, sending and

receiving data there between, and executing a desired voluntary data processing

process, said program transfer system comprising: a program storage means for storing

a controlling program for storing a plurality of functional module classes having a coded

processing logic, a controlling program for reading out at least one said functional

module classes when definition information is provided to the server computer and the

configuration information including at least functional module read-out information to

declare the contents of a data processing process to be executed and a processing

condition are sent from the server computer, and for dynamically generating a unit-

program processing by using the coded processing logic of said functional module

classes, and a controlling program for dynamically executing said unit-program of data

processing based on the processing condition included in said configuration information;

a program read-out means for reading out the controlling program from said program

storage means based on a request from at least one of the client computer and the data

processing server computer; and a transfer means for transferring said read-out

controlling program to at least one of the client computer and the data processing server

computer (see claim 1 and 10-13 rejections above).

As per claim 27, Yokote further teaches wherein said program storage means

stores a controlling program which stores at least one of the configuration information

corresponding to the data processing to be executed, said configuration information is

used for generating the unit-program processing, and reads out at least one of the   ·

configuration information based on a request reading out the configuration information

corresponding to the data processing to be executed when said request is sent from the

client computer (see claim 2 rejection above).

As per claim 28, Yokote teaches a program transfer system for transferring and

downloading a controlling program to at least one of a client computer and a data

processing server computer comprising a system for dynamically generating and

processing a program by connecting a server computer and at least one of the client

computer and the data processing server computer via a network means, sending and

receiving data there between, and executing a desired voluntary data processing process, said program transfer system comprising: a program storage means for storing a controlling program for storing a plurality of functional module classes having a coded processing logic, a controlling program for outputting a request for at least one of configuration information corresponding to a data processing to be executed, a controlling program for reading out at least one of said functional module classes when the configuration information including at least functional module at least read-out information and a processing condition are sent from the server computer and for dynamically generating a unit-program processing by using the coded processing logic of said functional module classes, and a controlling program for dynamically executing said unit-program processing based on the processing condition included in said configuration information; a program read-out means for reading out the controlling program from said program storage means based on a request from at least one of the client computer and the data processing server computer; and a transfer means for transferring said read-out controlling program to at least one of the client computer and the data processing server computer (see claim 1 and 10-13 rejections above).
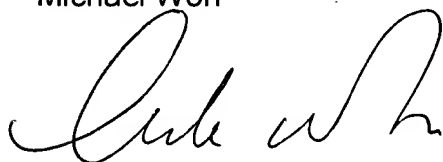
As per claim 29, Yokote further teaches wherein said server computer comprises said program storage means, said program read-out means and said transfer means (see claim 5 rejection above).

4.   Any inquiry concerning this communication or earlier communications from the

examiner should be directed to Michael Y Won whose telephone number is 571-272-

3993. The examiner can normally be reached on M-Th: 7AM-5PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's

supervisor, Hosain T Alam can be reached on 571-272-3978. The fax phone number

for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the

Patent Application Information Retrieval (PAIR) system. Status information for

published applications may be obtained from either Private PAIR or Public PAIR.

Status information for unpublished applications is available through Private PAIR only.

For more information about the PAIR system, see http://pair-direct.uspto.gov. Should

you have questions on access to the Private PAIR system, contact the Electronic

Business Center (EBC) at 866-217-9197 (toll-free).

Michael Won

January 6, 2005

HOSAIN ALAM
SUPERVISORY PATENT EXAMINER